

### Amendments to the Claims

1. (Currently Amended) A method for completion processing, comprising:
  - processing inbound TCP segments; and
  - performing completion processing of received TCP ACKS independently of the processing of the inbound TCP segments, wherein the processing of the inbound TCP segments continues without waiting for the completion processing of each received TCP ACK to finish.
2. (Original) The method of claim 1, wherein performing completion processing further comprises:
  - scheduling completion processing of each received TCP ACK using a completion coalescing list.
3. (Original) The method of claim 2, further comprising:
  - coalescing received TCP ACKS using the completion coalescing list.
4. (Currently Amended) ~~The method of claim 2, further comprising:~~ A method for completion processing, comprising:
  - processing inbound TCP segments;
  - performing completion processing of received TCP ACKS independently of the processing of the inbound TCP segments;
  - scheduling completion processing of each received TCP ACK using a completion coalescing list; and

passing completion information corresponding to each received TCP ACK to a completion handler via the completion coalescing list.

5. (Original) The method of claim 4, wherein the completion information corresponding to each received TCP ACK is passed to the completion handler via a connection context in the completion coalescing list.

6. (Original) The method of claim 5, further comprising:

chaining the connection context to the completion coalescing list.

7. (Original) The method of claim 6, wherein the completion context is only chained to the completion coalescing list if a connection corresponding to the connection context does not have a pending completion request.

8. (Original) The method of claim 7, wherein, if the connection corresponding to the connection context does have a pending completion request, updating information in the connection context.

9. (Original) The method of claim 8, wherein the updated information in the connection context comprises a last acknowledged sequence number (LastAckedSN).

10. (Original) The method of claim 4, further comprising:

selectively bypassing the completion coalescing list and sending the completion information directly to the completion handler for completion processing.

11. (Original) The method of claim 10, wherein selectively bypassing the completion coalescing list is performed in response to a retransmit operation.

12. (Original) The method of claim 2, further comprising:

processing completion of an RDMA Read Request upon reception of an RDMA Read Response using the completion coalescing list.

13. (Original) The method of claim 12, further comprising:

chaining a connection context corresponding to the RDMA Read Request to the completion coalescing list.

14. (Original) The method of claim 12, further comprising:

selectively bypassing the completion coalescing list and sending the RDMA Read Request directly to the completion handler for completion processing.

15. (Original) The method of claim 14, wherein selectively bypassing the completion coalescing list is performed in response to a Read Queue not having sufficient free space to place the RDMA Read Request.

16. (Original) A computer program product stored on a recordable medium, which when executed, performs the method set forth in claim 1.

17. (Currently Amended) A system for completion processing, comprising:  
TCP logic for processing inbound TCP segments; and  
a system for performing completion processing of received TCP ACKS independently of the processing of the inbound TCP segments, wherein the processing of the inbound TCP segments continues without waiting for the completion processing of each received TCP ACK to finish.

18. (Original) The system of claim 17, further comprising:  
a completion coalescing list for scheduling completion processing of each received TCP ACK.

19. (Original) The system of claim 18, wherein the completion coalescing list coalesces received TCP ACKS.

20. (Currently Amended) ~~The system of claim 18, further comprising:~~ A system for completion processing, comprising:

TCP logic for processing inbound TCP segments;

a system for performing completion processing of received TCP ACKS independently of the processing of the inbound TCP segments;

a completion coalescing list for scheduling completion processing of each received TCP ACK; and

a completion handler for completion processing of each received TCP ACK, wherein completion information corresponding to each received TCP ACK is passed to the completion handler via a connection context in a completion coalescing list.

21. (Original) The system of claim 20, wherein the TCP logic chains the connection context to the completion coalescing list.

22. (Original) The system of claim 21, wherein the TCP logic only chains the completion context to the completion coalescing list if a connection corresponding to the connection context does not have a pending completion request.

23. (Original) The system of claim 22, wherein, if the connection corresponding to the connection context does have a pending completion request, the TCP logic updates information in the connection context.

24. (Original) The system of claim 23, wherein the updated information in the connection context comprises a last acknowledged sequence number (LastAackedSN).

25. (Original) The system of claim 20, further comprising:

a transmit/retransmit handler for sending the completion information directly to the completion handler for completion processing, thereby selectively bypassing the completion coalescing list.

26. (Original) The system of claim 25, wherein the completion information is sent directly to the completion handler for completion processing by the transmit/retransmit handler in response to a retransmit operation requested by the TCP logic.

27. (Original) The system of claim 18, further comprising:

RDMA logic for handling received RDMA messages, and for scheduling an RDMA Read Request for completion processing by chaining a connection context corresponding to the RDMA Read Request to the completion coalescing list.

28. (Original) The system of claim 27, further comprising:

a completion handler for completion processing of each RDMA Read Request, wherein completion information corresponding to each RDMA Read Request is passed to the completion handler via the connection context in the completion coalescing list.

29. (Original) The system of claim 28, wherein the RDMA logic is configured to selectively bypass the completion coalescing list and send the RDMA Read Request directly to the completion handler for completion processing.

30. (Original) The system of claim 29, wherein the completion coalescing list is selectively bypassed in response to a Read Queue not having sufficient free space to place the RDMA Read Request.